WHAT IS CLAIMED IS:

1. A method of providing instructions to a processor from an emulation instruction register comprising:

receiving a plurality of instructions simultaneously from the emulation instruction register;

determining the validity of a first instruction of the plurality of instructions;

providing the first instruction of the plurality of instructions to a decoder if the first instruction is valid;

determining the validity of a second instruction of the plurality of instructions; and

providing the second instruction of the plurality of instructions to the decoder if the second instruction is valid.

2. The method of Claim 1, further comprising determining the size of the plurality of instructions.

3. The method of Claim 1, further comprising storing the plurality of instructions in a single instruction register.

4. The method of Claim 1, further comprising loading the second instruction of the plurality of instructions after determining the first instruction is invalid.

Dub A8

5. The method of Claim 1, further comprising loading the plurality of instructions in parallel into the emulation instruction register.

6. The method of Claim 1, further comprising providing the second instruction to the decoder after the first instruction is completed.

7. The method of Claim 1, further comprising providing the plurality of instructions to the decoder without receiving multiple RTIs.

8. The method of Claim 1, further comprising providing instructions to a digital signal processor.

9. A method of processing instructions within a processor comprising:

loading a plurality of instructions into a single instruction register;

receiving an RTI;

simultaneously providing the plurality of instructions to the processor; and

processing the plurality of instructions.

pub A8

10. The method of Claim 9, further comprising loading the plurality of instruction into an N-bit emulation instruction register.

11. The method of Claim 9, further comprising determining the validity of each of the plurality of instructions before processing.

12. The method of Claim 11, further comprising aborting the processing of any invalid instructions and loading a next instruction of the plurality of instructions.

13. The method of Claim 9, further comprising loading a next instruction of the plurality of instructions if a no-operation instruction is loaded.

14. The method of Claim 9, further comprising providing the plurality of instruction to the processor a plurality of times without reloading the instruction register.

15. The method of Claim 9, further comprising providing the plurality of instructions to a digital signal processor.

16. A processor comprising:

an instruction register adapted to store a plurality of instructions;

emulation control logic adapted to control the flow of the plurality of instructions to a processor pipeline following detection of a single RTI; and

a decoder which may receive the plurality of instructions for processing.

17. The processor of Claim 16, wherein the instruction register is an emulation instruction register.

18. The processor of Claim 16, wherein the control logic determines the validity of the plurality of instructions and discards any invalid instructions.

19. The processor of Claim 16, wherein the control logic loads a next instruction immediately after detecting a no-operation instruction.

20. The processor of Claim 16, wherein the processor is a digital signal processor.

21. An apparatus, including instructions residing on a machine-readable storage medium, for use in a machine system to handle a plurality of instructions, the instructions causing the machine to:

load the plurality of instructions into a single instruction register;

receive and RTI;

provide the plurality of instructions to the processor; and

process the plurality of instructions.

22. The apparatus of Claim 21, wherein the instruction register is an emulation instruction register.

23. The apparatus of Claim 21, wherein the validity of each of the plurality of instructions is determined before processing.